**Evaluation**

My application fulfils all of its functions, which were to keep a record of employees and their tasks and calculate their pay.

It does not have the features the client requested – import and export – although it does have a save feature.

I deemed the save feature to be a function rather than a feature, as one of the goals of the application was to be an improvement over a spreadsheet, and without a save function this wouldn’t be possible, regardless of how many other features it had.

I thought that the import feature was unnecessary, as this is a bespoke application to be used by one person, so the feature would only ever be used once, and therefore would not be worth the time it would take to implement.

I also did not add an export feature, due to time constraints and the complexity of implementing it. However, data can be shared between instances of the application by manually copying the data folder and putting in the folder of another application, a happy side-effect of the way the data is stored. This means that it is relatively easy to transfer all the saved data between different versions of the program, as my client tried a few different updated versions while I was testing it.

Another unrequested feature I added was the option to assign a picture to an employee, to make it clear which one is selected at a glance. The picture can be changed by clicking on it, which is good for ease of use.

The UI is very plain, but the layout is primarily for making it easy to see information, which it does well.  
The controls are clumsy, with employees being added via the menu, but editing and adding tasks being done by buttons – also the employee/task must be selected before editing/assigning a task, which may confuse the average user.  
However, as this a bespoke application just for my client, I simply told to use the ‘help’ feature I included, which opens a HTML document in the default browser, fully documenting how to use the program, and after reading the document my client said the program was simple to use.

The code is not particularly tidy, efficient or well written, as I was primarily concerned with making it functional first. Rather than making procedures for near-identical functions I copied and pasted code between different parts of the program. I also re-used variable names between similar algorithms, such as the ones for generating file paths to the task data. This will make the code difficult to maintain or edit, as it is not clear they are the same variables. Some variables also have very similar names, and it is not always clear exactly what they are for, particularly those in loops and the names of WinForm elements.  
I have commented the code extensively to make it clear to myself and anyone else who may use it to make it clear what each part does, which somewhat negates the problem of the untidiness and bad variable naming.

Given more time, I would first tidy up the code. This would include making procedures for things such as generating file paths and updating labels, and giving the variables clearer names, particularly the WinForm elements.  
I would implement an export feature, and perhaps an import feature if my client still wanted it.